International Journal of Computer Science Engineering
and Information Technology Research (IJCSEITR)
ISSN(P): 2249-6831; ISSN(E): 2249-7943
Vol. 4, Issue 4, Aug 2014, 121-126

DESIGN AND IMPLEMENTATION OF DYADIC SHIFT AND GRAPHICAL INVERSE OF STRING SEQUENCES USING REWRITING CYCLIC NORMAL AUTOMATON

U. M. FERNANDES DIMLO[1] & E. G. RAJAN[2]

[1]Research Scholar, University of Mysore, Karnataka, India

[2]Founder & President, Pentagram Research Center Pvt. LTD, Hyderabad, Andhra Pradesh, India

ABSTRACT

Algorithms that combine numeric and symbolic computing have been of increasing importance and interest over the past decade. The necessity to work with noisy and imprecise data and for speed and accuracy with algebraic and numerical problems encourages the synergy between two types of computing such as numeric and symbolic. In this paper signals are represented as strings or words which can be stored in computer memory, further normal algorithms are written to perform operations like dyadic shift and graphical inverse signals using rewriting cyclic normal automata.

KEYWORDS: RCNA, Normal Algorithms, Substitution Formula

INTRODUCTION

Let A be an alphabet and A* is free monoid. Every algorithm over A have equivalent normal algorithm. Any algorithm with set of inputs and outputs can be transformed [1] into normal algorithm with same set of input and output pair. This can be realized using normal automata and as well as with family of automata. Every family of automata over a potentially realizable alphabet is normalisable and therefore it is impossible to construct a non normalisable automaton.

Characteristics of family of automata are as follows

- Only basic arithmetic operations are used to accept any input letter.
- Memory used is unbounded which is under the control of automaton [8]
- The set of words which reach any one state of F from initial state are part of output.
- The input data is stored is internal memory where backtracking is not allowed.
- The acceptance or rejection of any word is depends upon transition rules [9].

DEFINITION OF IMPORTANT TERMS

Rewriting Cyclic Normal Automata

Rewriting cyclic normal automata is triple as <Q, INI, FIN> with 4 type of edges where Q is set of all states, INI is initial state and FIN is final states [12].

Normal Algorithm

Normal algorithm consists of set of productions or substitution formulas [15] either simple or terminal type to generate given word in A* where A is an alphabet and A* is free monoid.

Convolution

Convolution is mathematical [14] technique which relates the output y[n] linear time variant system to input x[n]. Linear convolution is the process of multiplying any two given non integer sequences of signals.

Dyadic Shift

In dyadic shift given word has been divided into two parts and shifted such that second part begins the string where first part ends the string.

Substitution Formula

The formula of the form P->Q used in normal algorithms to derive given string is called as substitution formula. There are two types of substitution formulas [13]

- Simple substitution formula
- Terminal substitution formula

This formula comprises of generic variables, auxiliary variables and string. Terminal formula will be used to end the derivation where simple formula used to replace part of a string by another string while derivation.

Generic Variables

Variable whose values are taken from the letters [6] of the alphabet are known as generic variable. Generic variables are denoted by the symbolsξ, η andμ.

DYADIC SHIFT ON SIGNALS

Let w be a string from A* where A is an alphabet. The dyadic shift is nothing but [10] dividing the given string into two equal parts and interchanging them in the position. Let A = {w, x, y, z} and w=wxyz then after dyadic shift w=yzxy. Let us assume that strings are of even length hence we can divide them into equal parts.

Dyadic shift can be accomplished using left shift n/2 times where n is length of the given string which is of even number. For example, consider the alphabet A = {w, x, y, z} and the auxiliary alphabet E = {$\alpha\ \delta\,\Gamma$}. Let μ and ξ be the generic variables that range over A, and let w = wxyz be a word from A. Normal algorithm for left shift is as follows

Table 1: Formulas for Dyadic Shift

Formula	Formula Number	Comments
$\mu\xi\Gamma \to \xi\Gamma\mu$	0	Move generic variable μ rightmost of string.
$\alpha\mu \to \mu\alpha$	1	Move α from left to right of string
$\alpha \to \Gamma$	2	Replace α by Γ
$\Gamma \to \cdot$	3	Move Γ right to left using dot
$\to \alpha$	4	Introduce α at head of given string

The process of finding dyadic shift of abcd will be completed by applying rule 4, rule 1, rule 2 and rule 3 successively for 2 times because number of characters in a given string is four.

Table 2: Dyadic Shift of Word Wxyz

Step No.	Transformation	Formula Used
1	wxyz	Input string
2	αwxyz	4
3	wαxyz	1
4	wxαyz	1
5	wxyαz	1
6	wxyzα	1
7	wxyzΓ	2
8	wxzΓy	3
9	wzΓxy	3
10	zΓwxy	3
11	zwxy	3
12	αzwxy	4
13	zαwxy	1
14	zwαxy	1
15	zwxαy	1
16	zwxyα	1
17	zwxyΓ	2
18	zwyΓx	3
19	zyΓwx	3
20	yΓzwx	3
21	yzwx	Resultant string

The RCNA for the above EPT system is defined by C^{DS}= <Q,INI,FIN> where$Q=\{q_{00},q_{10},q_{20},q_{11},q_{12},q_{30},q_{21},q_{22},q_{40},q_{31},q_{32},q_{50},q_{41},q_{42},q_{43},q_{44},q_{60},q_{61}\}$ the set of edges T where, T =

$(q_{INI}, \Sigma^*, q_{00})$,	$(q_{00}, \Sigma\backslash\xi/\Leftarrow, q_{10})$,	$(q_{10}, \Sigma\backslash\xi/\Leftarrow, q_{20})$,	$(q_{10}, /\rightarrow\alpha, q_{11})$,
$(q_{11}, \mu/\rightarrow, q_{12})$,	$(q_{12}, \Lambda/\vdash, q_{00})$,	$(q_{20}, \Sigma\backslash\xi/\Leftarrow, q_{30})$,	$(q_{20}, \alpha\mu/\mu\alpha, q_{21})$,
$(q_{21}, \mu/\rightarrow, q_{22})$,	$(q_{22}, \Lambda/\vdash, q_{00})$,	$(q_{30}, \Sigma\backslash\xi/\Leftarrow, q_{40})$,	$(q_{30}, \alpha/\Gamma, q_{31})$,
$(q_{31}, \mu/\rightarrow, q_{32})$	,$(q_{32}, \Lambda/\vdash, q_{00})$,	$(q_{40}, \Sigma\backslash\xi/\Leftarrow, q_{50})$,	$(q_{40}, \mu\beta\Gamma/\ \beta\Gamma\mu, q_{41})$
$(q_{41}, /\rightarrow\alpha, q_{42})$,	$(q_{42}, \alpha\mu/\mu\alpha, q_{43})$,	$(q_{43}, \mu/\rightarrow, q_{44})$,	$(q_{44}, \Lambda/\vdash, q_{00})$,
$(q_{50}, \Sigma\backslash\xi/\Leftarrow, q_{60})$,	$(q_{50}, \mu\beta\Gamma/\ \beta\Gamma\mu, q_{FIN})$,	$(q_{60}, \mu/\rightarrow, q_{61})$,	$(q_{61}, \Lambda/\vdash, q_{00})$ }

The pictorial representation of dyadic shift using RCNA is shown in figure 1

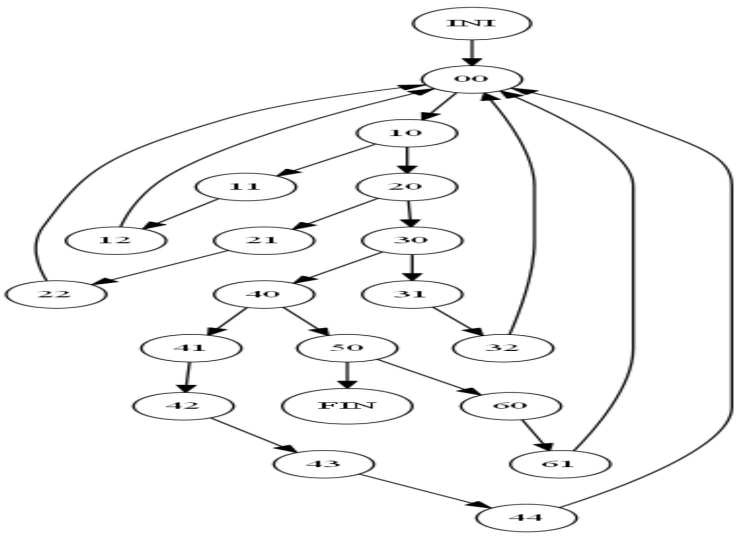

Figure 1: RCNA Diagram for Dyadic Shifting

GRAPHICAL INVERSE OF GIVEN SEQUENCE

Let A be an alphabet and A* be free monoid which contains set of all strings generated using symbols in A including null string. Let w be any string from A* to which we have to find graphical inverse. Inverse of a given string can be found by performing left a shift n time where n is length of the given string.

For example, consider the alphabet A = {a b c d}, and the auxiliary alphabet E = {α δ Γ}. Let μ and ξ be the generic variables that range over A, and let w = abcd be a word from A. Normal algorithm for left shift is as follows

Table 3: Substitution Formulas for Inverse

Formula	Formula Number	Comments
$\mu\xi\Gamma \to \xi\Gamma\mu$	0	Move generic variable μ rightmost of string.
$\alpha\mu \to \mu\alpha$	1	Move α from left to right of string
$\alpha \to \Gamma$	2	Replace α by Γ
$\Gamma \to \cdot$	3	Move Γ right to left using dot
$\to \alpha$	4	Introduce α at head of given string

The process of finding reverse of abcd will be completed by applying rule 4, rule 1, rule 2 and rule 3 successively for 4 times because number of characters in a given string is four.

Table 4: Graphic Inverse of String Abcd

Step No.	Transformation	Formula Used
1	abcd	Input string
2	αabcd	4
3	aαbcd	1
4	abαcd	1
5	abcαd	1
6	abcdα	1
7	abcdΓ	2
8	abdΓc	3
9	adΓbc	3
10	dΓabc	3
11	dabc	3
12	αdabc	4
13	dαabc	1
14	daαbc	1
15	dabαc	1
16	dabcα	1
17	dabcΓ	2
18	dacΓb	3
19	dcΓab	3
20	dcab	3
21	αdcab	4
22	dαcab	1
23	dcαab	1
24	dcaαb	1
25	dcabα	1
26	dcabΓ	2
27	dcΓba	3
28	dcba	Resultant string

The RCNA for the above EPT system [11] is defined by C^{GI}= <Q,INI,FIN> where $Q=\{q_{00},q_{10},q_{20},q_{11},q_{12},q_{30},q_{21},q_{22},q_{40},q_{31},q_{32},q_{50},q_{41},q_{42},q_{43},q_{44},q_{60},q_{61},q_{51},q_{52},q_{53},q_{54},q_{70},q_{71}\}$ the set of edges T where, T=

$\{(q_{INI}, \Sigma^*, q_{00})$,	$(q_{00}, \Sigma\backslash\xi/\Leftarrow,q_{10})$,	$(q_{10},\Sigma\backslash\xi/\Leftarrow,q_{20})$,	$(q_{10},/\rightarrow\alpha,q_{11})$,
$(q_{11},\mu/\rightarrow,q_{12})$,	$(q_{12}, \Lambda/\vdash,q_{00})$,	$(q_{20},\Sigma\backslash\xi/\Leftarrow,q_{30})$,	$(q_{20}, \alpha\mu/\mu\alpha,q_{21})$,
$(q_{21},\mu/\rightarrow,q_{22})$,	$(q_{22}, \Lambda/\vdash,q_{00})$,	$(q_{30},\Sigma\backslash\xi/\Leftarrow,q_{40})$,	$(q_{30}, \alpha/\Gamma,q_{31})$,
$(q_{31},\mu/\rightarrow,q_{32})$,	$(q_{32}, \Lambda/\vdash,q_{00})$,	$(q_{40},\Sigma\backslash\xi/\Leftarrow,q_{50})$,	$(q_{40},\mu\beta\Gamma/ \beta\Gamma\mu,q_{41})$,
$(q_{41},/\rightarrow\alpha,q_{42})$,	$(q_{42}, \alpha\mu/\mu\alpha,q_{43})$,	$(q_{43},\mu/\rightarrow,q_{44})$,	$(q_{44}, \Lambda/\vdash,q_{00})$,
$(q_{50},\Sigma\backslash\xi/\Leftarrow,q_{60})$,	$(q_{50}, \mu\beta\Gamma/ \beta\Gamma\mu,q_{51})$,	$(q_{51},/\rightarrow\alpha,q_{52})$,	$(q_{52}, \alpha\mu/\mu\alpha,q_{53})$,
$(q_{53},\mu/\rightarrow,q_{54})$,	$(q_{54}, \Lambda/\vdash,q_{00})$,	$(q_{60},\Sigma\backslash\xi/\Leftarrow,q_{70})$,	$(q_{60}, \mu\beta\Gamma/ \beta\Gamma\mu,q_{61})$,
$(q_{61}, \Gamma/\Lambda,q_{FIN})$,	$(q_{70},\mu/\rightarrow,q_{71})$,	$(q_{71}, \Lambda/\vdash,q_{00})\}$	

The graphical representation to find inverse is shown figure 2.

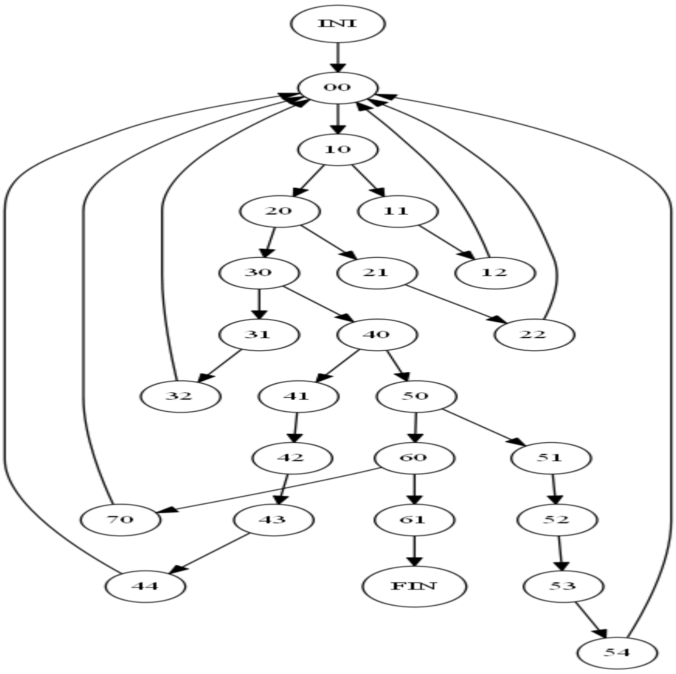

Figure 2: Graphical Inverse Using RCNA

CONCLUSIONS

In this paper a new technique for performing operations on signals using normal algorithms and RCNA has been represented, which solves the problems like numeric computation of signals, representation of signals in computer process able format. The use of automata in signal processing is still in infancy stage. More efforts are required to explore various methods using automata to implement various signal processing operations. The features of automata will make them as alternative for designing signal operations as conventional methods. In summary, it is considered that RCNA based automata have an excellent potential in signal processing.

REFERENCES

1. Lothaire, M., Combinatorics on Words, Addison Wesley Pub. Co., 1983.
2. Markov, A.A., Theory of Algorithms, The Isreal Program for Scientific Translations, 1961.

3. Mittelstadt, Peter, Quantum Logic, D. Reidel Publishing Company.

4. Pin, J.E., Varieties of Formal Languages, Plenum Press, New York, 1986.

5. Shanin, N.A., On Constructive Real Numbers and Constructive Function Spaces, AMS, providence, Rhode Island, 1968.

6. Shanin, N.A., On the Constructive Interpretation of Mathematical Judgements, AMS, Translations 23, 1963.

7. Toffoli, Tommaso, and Margolus, Norman, Cellular Automata Machines, The MIT Press, Cambridge, Massachusetts, 1987.

8. Wolfram, Stephen, Theory and Applications of Cellular Automata, World Scientific Publishing Company, 1986.

9. Birkhoff, G., and Von Neumann, The Logic of Quantum Mechanics, Annals of Mathematics, Vol 37, No.4, 1936.

10. Finch, P.D., On the Structural of Quantum Logic, Jr. of Symbolic Logic Vol. 34, No.2, 1969.

11. Rajan, E.G., An Elementary Formal System for representing constructive systems, World Multiconference on Systemics, Cybernetics and Informatics, Caracas, Venezuela, 1997.

12. Rajan, E.G., Special Automata for Normal Algorithms and their Transcriptions, World Multiconference on Systemics, Cybernetics and Informatics, Caracas, Venezuela, July 1997.

13. Rajan, E.G., MEDIMAGE - A PC Based Medical Image Understanding System, invited paper, National Conference, Model College of Engineering, Cochin, organized by All India Council of Technical Education, Government of India, 1997.

14. Rajan, E.G., Fast Algorithm for collision avoidance by autonomous mobile systems in the framework of cellular logic array processing, 13th international conference on CAD/CAM Robotics & Factories of the Future, Universidad Technological de Pereira, Columbia, South America, December 15-17, 1997.

15. Scott, D., Suppes, P., Foundational Aspects of the Theory of Measurement, Jr. Symbolic Logic, Vol. 23, No.2, January 1958.